

Agentic Sales

“Schweizer KMU”


Demo Use Case

Marc Gloor
<marc.gloor@u.nus.edu>

AUFGABE

Owner AI Sales Agent / Sales-Tools



Bei  unterstützen wir Schweizer KMU dabei, in der digitalen Welt erfolgreich zu sein. Stell dir folgende reale Situation aus dem Sales-Alltag vor:

Ein Sales Manager kommt mit zwei Excel-Listen zu dir:

- **Liste A (aus Salesforce):** Kundennummer, Kundenname, Telefon, E-Mail, Sprache, gekaufte Produkte, Umsatz, Vertragsablaufdatum u.a.
- **Liste B (öffentliche Daten von local.ch):** Website des Kunden, Name des Standorts, Kundennummer.

Sein Ziel:

1. Kunden, die **kurz vor Vertragsablauf** stehen, sollen mit einer **Promotionsaktion** reaktiviert werden.
2. Bei den **übrigen Kunden** möchte er prüfen, ob sich ihnen eine **neue Website** verkaufen lässt – dazu sollen die **digitalen Potenziale auf ihrer bestehenden Website** analysiert werden.

Deine Aufgabe

Beschreibe, wie du diese Aufgabe angehen würdest. Als konkretes Ergebnis möchten wir den oder die **Prompts** sehen, die du an das/die von dir gewählte(n) Tool(s) geben würdest – also das, was du tatsächlich eintippen würdest, um zum Ziel zu kommen.

Bitte liefere uns:

1. **Den/die Prompt(s)**, ausformuliert, so wie du sie einsetzen würdest.
2. **Eine kurze Begründung** (max. ½ Seite): Welche(s) Tool(s) hast du gewählt und warum? Welche Arbeitsschritte trennst du voneinander und weshalb?

Rahmen

- Halte dich insgesamt kurz. Uns interessiert dein Denken, nicht eine fertig gebaute Lösung.
- Es gibt **keine einzig richtige Antwort** und kein „richtiges“ Tool. Wir bewerten, *wie* du das Problem zerlegst und *warum* du dich für deinen Weg entscheidest.
- Du darfst Annahmen treffen, wo Informationen fehlen – mach sie einfach transparent.

Vorgehensweise

Jede einzelne Zeile des folgenden Prompts wurde von mir manuell **handgeschrieben ohne LLM Involvement**.

Single-Agent / Multi-Agent

Aus Gewohnheit habe ich den Prompt in Englisch geschrieben. Der Einfachheit halber habe ich den Workflow als **Single-Agent-Pipeline** konstruiert, eine Multiagent-Pipeline wäre für diese relativ einfache Aufgabenstellung ein Overhead.

Wahl von Agent / Provider / Model

Ich habe mich für einen lokal installierten HERMES Agent (unter Debian Linux) und aus Privacygründen das lokale Llama 4 Scout Model mit 40b Parametern entschieden welches lokal über GPU's betrieben wird damit keine sensiblen Kundendaten über den Provider verarbeitet werden. HERMES hat ein hervorragendes Memory System, ist selbstlernend und robust. Das gewählte Modell muss zudem stark genug sein um Toolcalls sauber zu verarbeiten. Zusätzlich muss im LAN sichergestellt werden, dass das betreffende HERMES gateway keine Port connections zulässt und hinter einer Firewall gesichert ist.

Prompt Context Engineering

Für Agentic Prompts benutze ich üblicherweise den '**Context Engineering**'-Approach, ein Ansatz welcher vom bekannten Entwickler Karpathy beschrieben wurde und die eigentliche Kernkompetenz für ernsthafte KI-Programmierung darstellt.

Gemäss Karpathy leiden LLMs unter „Aufmerksamkeitsverdünnung“, wenn sie lange, kreative und dialogbasierte Prompts erhalten. Ein strukturierter Prompt-Layout soll mithilfe von sauberem Markdown und XML-Tags definiert werden um die semantische Gewichtung der Token zu maximieren und die Transformation von kreativen Text in eine zuverlässige agentenbasierte Zustandsmaschinen-Pipeline gewährleisten.

Prompt Architektur

Innerhalb der **GENERAL_INSTRUCTION** habe ich die Erwartungen für eine geplante Prompt Review gestellt. In den **SYSTEM_DIRECTIVES** wird die Identität des Agents, die Vorgehensweise der Abarbeitung und den Befehl, einen Skill aus der Prompt zu erstellen, gesetzt.

Die **SYSTEM_VARIABLES** definieren den Workspace, Pfadangaben und Formate.

Phase 0 ist ein Filesystem clean-up im Workspace.

Phase 1 parst die beiden XLS-Files und kompiliert daraus eine merged CSV Liste.

In **Phase 2** werden zwei spezifische CSV-Kampagnen-Listen aus der merged Kundenliste generiert.

In **Phase 3** werden die E-Mail Kampagnen als DRAFT angelegt welche an die Kunden adressiert sind, deren Lizenzprodukte im laufenden Quartal verfallen. Hermes verwendet ein im Vorfeld vorbereitetes E-Mail Template.

In **Phase 4** werden die Websites der restlichen Kunden auf alte Inhalte geparkt und darauf eine E-Mail Kampagne analog Phase 3 assembliert. Wie HERMES 'merkt' das die Website alt ist, steht unten im Code!

Phase 3 und Phase 4 sind Pseudocode und müssten noch spezifischer ausformuliert und **mit künstlichen Testdaten agentic getestet werden** bis das Endresultat plausible Resultate erzeugen.

<GENERAL_INSTRUCTION>

-Generate a structured prompt layout uses clean markdown and XML delimiters to maximize semantic token weight.
-Transform the following creative text into a reliable agentic state-machine pipeline.
-Create a phased context based approach that includes quality gates that have to be approved manually by at the console.

</GENERAL_INSTRUCTION>

<SYSTEM_DIRECTIVES>

-You are an expert Sales Agent. Define the role of a Sales Agent for Small an Medium Enterprises. Reflect his quote carrying responsibility and ensure the following goals are reflected accoringly: ARR (Annual Recurring Revenue), IRR (Internal Rate of Return) and TCR (Total Contract Revenue).

-Execute the subsequent phases sequentially.

-Prioritize precision and strict adherence to the constraints.

-Skills: Memorize this skill and name it "DeineFirma-contract-and-website-renewal-campaign" so it can be re-applied.

</SYSTEM_DIRECTIVES>

<SESSION_VARIABLES>

WORKSPACE_ROOT="./.hermes/workspace/2026-06_sales-campaign"

INPUT_XLS_FILE_A="<WORKSPACE_ROOT>/inputfiles/salesliste-A.xls"

INPUT_XLS_FILE_B="<WORKSPACE_ROOT>/inputfiles/salesliste-B.xls"

SALES_SOURCE_LIST_FORMAT_A="Kundennummer;Kundenname;Telefon;E-Mail;Sprache;Produkt;Umsatz;Vertragsablaufdatum;<Weitere Felder>;..."

SALES_SOURCE_LIST_FORMAT_B="Kundennummer;Webseite;Standort"

SALES_TARGET_LIST_MERGED="<WORKSPACE_ROOT>/<YYYYMMDD-HHMM_Sales-List.csv>"

SALES_PROMO_CANDIDATES="<WORKSPACE_ROOT>/<YYYYMMDD-HHMM_Sales-Promo-Candidates.csv>"

SALES_WEBSITE_CANDIDATES="<WORKSPACE_ROOT>/<YYYYMMDD-HHMM_Sales-Website-Candidates.csv>"

</SESSION_VARIABLES>

<PHASE_0_INITIALIZATION>

-Move (mv) previously assembled SALES_TARGET_LIST_MERGED csv files to "<WORKSPACE_ROOT>/old-campaigns/"

</PHASE_0_INITIALIZATION>

<PHASE_1_DATABASE_BUILD>

-Export INPUT_XLS_FILE_A into plaintext ASCII colon separated files (csv) and store it in <WORKSPACE_ROOT> as salesliste-A.csv

-Export INPUT_XLS_FILE_B into plaintext ASCII colon separated files (csv) and store it in <WORKSPACE_ROOT> as salesliste-B.csv

-You have to compile a new file named <SALES_TARGET_LIST_MERGED> which consists of the two extracted csv lists where its unique primary-key is the Customer-ID "Kundennummer". Do only populate fields given in

<SALES_SOURCE_LIST_FORMAT_A> and <SALES_SOURCE_LIST_FORMAT_B>.

<SALES_SOURCE_LIST_FORMAT_A> and <SALES_SOURCE_LIST_FORMAT_B>.

</PHASE_1_DATABASE_BUILD>

<PHASE_2_BUILD_PROMO_AND_WEBSITE_CAMPAIGN_LISTS>

-Compile the new file named <SALES_PROMO_CANDIDATES> using following instructions:

SORT (ASCENDING) every line of the file <SALES_TARGET_LIST_MERGED> using the primary key "Kundennummer" and

FILTER only line items WHERE the field 'Vertragsablaufdatum' is shorter or equal than a calenar quarter (3

months). Compile a 2nd file named <SALES_WEBSITE_CANDIDATES> which reflects the delta of

<SALES_TARGET_LIST_MERGED> minus <SALES_PROMO_CANDIDATES>.

-Crosscheck: The sum of line numbers from <SALES_PROMO_CANDIDATES> plus (+) <SALES_WEBSITE_CANDIDATES> must equal

the total line number of <SALES_TARGET_LIST_MERGED>. If not, redo <PHASE_2_BUID_PROMO_AND_WEBSITE_TARGETS> until

the line numbers match, alternatively stop the script with the error message "Target Compilation of the sales

list failed...".

</PHASE_2_BUILD_PROMO_AND_WEBSITE_CAMPAIGN_LISTS>

This is the 1st work-horse part, only pseudo-code for now / need to be developed & tested properly

<PHASE_3_RUN_PROMO_CAMPAIGN>

-Create one e-mail per line item from <SALES_PROMO_CANDIDATES> and populate it as follows:

TO: 'E-Mail'

SUBJECT: Special Promotion: Your <Produkt> is expiring soon.

BODY: Take the e-mail template file (.msg or .eml) stored under

ATTACHEMENT: Attached the provioied special-promo offer pdf file.

-Save each assembled e-mail in the Mail clients draft folder.

</PHASE_3_RUN_PROMO_CAMPAIGN>

This is the 2nd work-horse part, only pseudo-code for now / need to be developed & tested properly

<PHASE_4_RUN_WEBSITE_CAMPAIGN>

-Check each line item from <SALES_WEBSITE_CANDIDATES> and use the web skill to analyze their URL from the field 'Webseite'.

-The /goal is to find outdated websites, either technically or content wise, use the following indications to qualify them as "outdated":

-Find outdated copyrights, not matching the current year (2026)

-Hunting for HTTP. Modern websites exclusively use HTTPS nowadays.

-Use 3rd party service (<TO-BE-DEFINED>) to idenity websites that are not mobile phone compliant.

-Search for "Powered by Wordpress" or "Powered by Joomla" or equal tags which indicates old websites.

-Search for cgi scripts ending with .cgi. Websites using CGI are outdated and not state of the art anymore.

-Do more research and find more nostalgic indications highlightn the website is out of date or old.

-Export qualified customers to dedicated file in the <WORKSPACE_ROOT>

-Write an e-mail draft campaign simliar to PHASE_3.

</PHASE_4_RUN_WEBSITE_CAMPAIGN>